Entropy: This term is used herein to refer to a well-known measure for a probability distribution.

Insertion operation: This term is used herein to refer to a process of inserting or adding a component of length one or more to a password, or adding a character to a component.

Known password or Real-user password: As used herein, these terms are used interchangeably to refer to a word or key that has become known to the public intentionally or unintentionally.

Length: This term is used herein to refer to the number of adjacent characters of the same type. For example, if length of a numerical component is one (1), then the component contains one number that has no numbers adjacent to it.

Letter: This term is used herein to refer to one or any sequence or combination of alphabetic characters. For example, letters within the English language may include letters A to Z.

Limited modification: This term is used herein to refer to a minimal change to a password, such that the modified password or second proposed password is still memorable to a user. A limited modification typically modifies a password with limited edit distance, which refers to the number of components or characters in modification (i.e., inserted, deleted, substituted, etc.) within a password. Typically, an edit distance of only one provides a password with sufficient strength, but greater than edit distance one is contemplated as well. The edit distance is "limited" in that only modification is allowed that would allow the modified password or second proposed password to remain memorable for the user

Password guess: This term is used herein to refer to a password generated that has an associated probability value.

Probabilistic context-free grammar: This term is used herein to refer to a common notion of a grammar generated through the training step of the current invention by learning base structure and component structure probabilities.

Probabilistic password cracking system: This term is used herein to refer to a methodology and model of effectively and efficiently attempting to crack a password through the use of probability values assigned to the password guesses or to structures associated with the password guesses. The probabilistic password cracking system generates guesses in highest probability order based on the training it received.

Probability distribution: This term is used herein to refer to application of passwords that might be generated by probabilistic grammar or the expected distribution in the wild. Thus, if a system remains updated with the most recent known words, distribution of passwords and probability values can remain accurate.

Proposed password: This term is used herein to refer to sequence or combination of alpha, numeric and/or special characters that is inputted by a user or generated by the system, and is subject to cracking by password cracking systems and modification by the current password analyze-modify system.

Relevant password distribution: This term is used herein to refer to the distribution induced or represented by the probabilistic context-free grammar.

Real-user password: This term is used herein to refer to a word or key, possibly referring to a password, that has become known to the public either intentionally or unintentionally

Special character: This term is used herein to refer to any sequence or combination of non-alpha and non-digit symbols. For example, non-alpha and non-digit symbols may include !@#$%^&*( )-_=+[ ]{ };':",./<>?.

Substitution operation: This term is used herein to refer to a process of substituting or exchanging one character with another character.

Sufficient complexity: This term is used herein to refer to a password having a strength or effectiveness that meets or surpasses the threshold complexity value.

Threshold complexity value: This term is used herein to refer to a quantitative point at which a password is deemed strong or weak. If a password satisfies the threshold complexity value, for example by requiring a large amount of guesses over a period of time to be cracked, then the password is deemed strong. If the passwords fails to meet the threshold complexity value (i.e., is too easy to crack), then the password is deemed weak.

Transposition operation: This term is used herein to refer to a process of exchanging two adjacent components.

Type: This term is used herein to refer to a grouping of one or more characters within a component. Examples of types include alphabetic characters, numerical digits, and special characters.

User: This term is used herein to refer to an individual attempting to test the current password analyze-modify system or attempting to develop a password for a secure account that requires authentication.

The advantages set forth above, and those made apparent from the foregoing disclosure, are efficiently attained. Since certain changes may be made in the above construction without departing from the scope of the invention, it is intended that all matters contained in the foregoing description or shown in the accompanying drawings shall be interpreted as illustrative and not in a limiting sense.

It is also to be understood that the following claims are intended to cover all of the generic and specific features of the invention herein disclosed, and all statements of the scope of the invention that, as a matter of language, might be said to fall therebetween.

What is claimed is:

1. A computer-implemented method of analyzing and modifying a first proposed password chosen by a user for a secured user account, said method comprising the steps of:

generating a probabilistic context-free grammar from an array of control passwords aggregated from real-user passwords;

establishing a threshold complexity value based on effort required to crack said array of control passwords,

said first proposed password including a base structure containing a plurality of components, wherein the step of establishing said threshold complexity value includes

setting a lower bound for a number of password guesses for said first proposed password until said threshold complexity value is reached, wherein said password guesses do not need to be generated,

estimating a number of components in said base structure that are greater than said threshold complexity value, and

estimating and establishing said threshold complexity value based on the forgoing steps;

receiving said first proposed password as inputted by said user into a computer interface of a computer system connected to a network;

deriving a complexity value of said first proposed password based on said context-free grammar;

comparing said complexity value of said first proposed password and said threshold complexity value,